Using OAuth with WRT Widgets

What is it for

You give someone limited access to your application with a special key, while using your regular key to unlock everything.

General

OAuth is an open protocol to allow secure API authorization in a simple and standard method from desktop and web applications. Quick introduction and more detailed information from it can be found from: oauth.net.

Example

HelloOAuth.zip shows three simple use cases for utilizing OAuth with an OAuth test server located at term.ie.

Use case 1: Request token

With this use case only the consumer key and secret are used for requesting an access token. Thus no other arguments are required to be supplied. Note that there is also additional test case, where incorrect secret is used for checking the server reply on error situations.

The request token would then be used in second use case where an access token is requested

Use case 2: Access token

With the example server, any API usage would require an access token, and for access token you would need to get the request token (with use case 1 codes). Note that for simplicity, the actual requests token and secret are hard-coded with the example.

The basic difference between use case 1 and 2 is that now, as there are two additional variables (request token & token secret), you need to add correct token Secret to the accessors used as an second argument for constructing the signature. Also you need to add the token into the parameters of the URL message.

The access token fetched could then be used with use case 3 for accessing the echo service.

Use case 3: Echo service

The OAuth test server used with this example only has one service, which is echo service that echoes back all non-OAuth parameters. Utilizing this service requires access token and secret for it, for simplicity, the actual access token and secret are hard-coded with the example.

The access token and secret are used in same places as the request token and secret were used with use case 2. The only real difference with this use case when compared to the use case 2 is that with this use case there are additional arguments supplied with the request.

The additional arguments must be added into the parameters of the URL message, so the arguments are also used when calculating the signature, adding anything to the URL after the signature has been made will invalidate the signature.